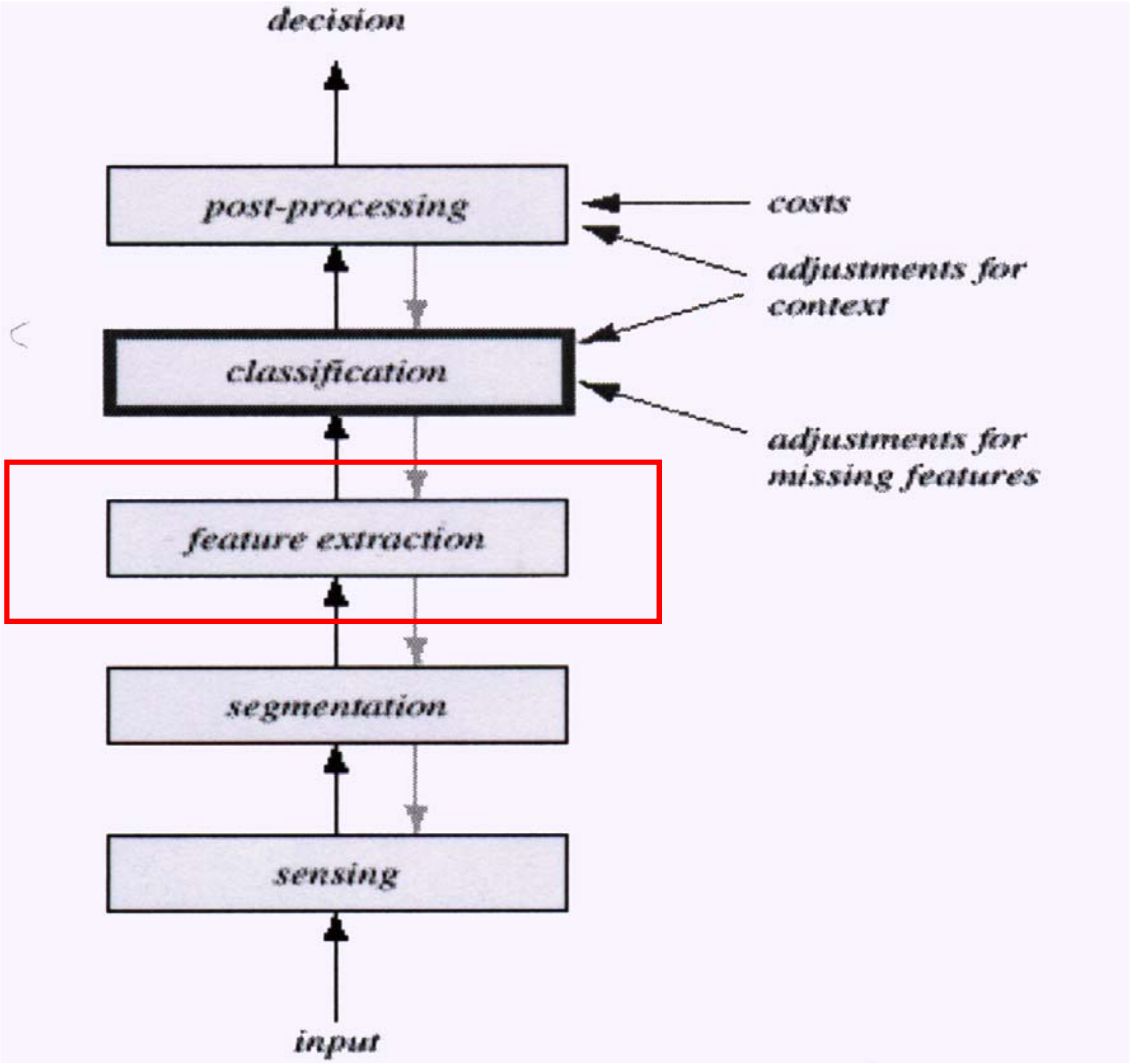
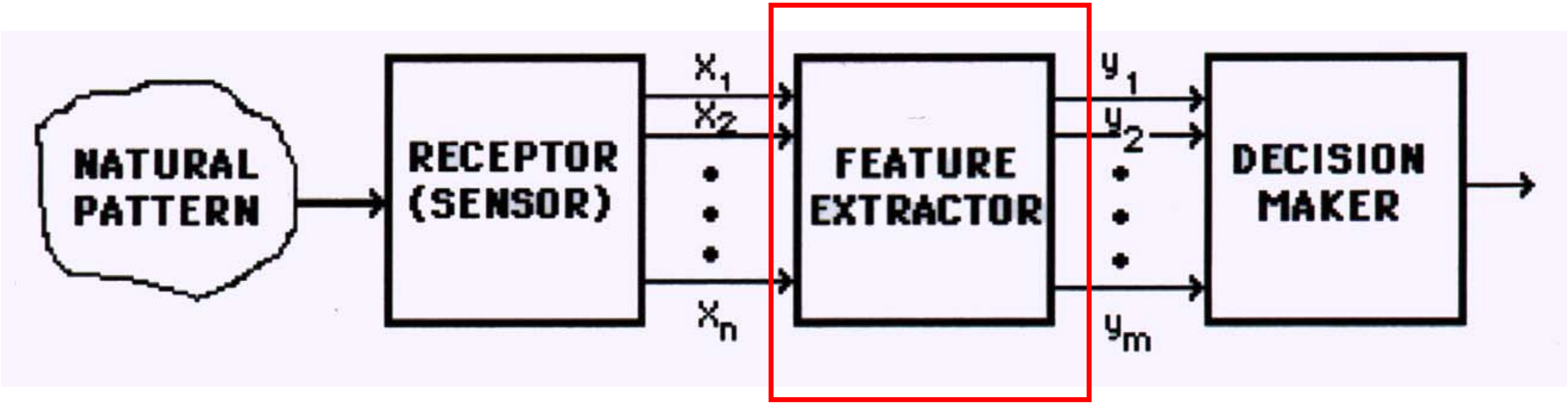
7. Feature Extraction for Representation and Classification

Pattern Recognition Systems



Pattern Recognition Systems

- **Feature extraction**
  - Discriminative features
  - Invariant features with respect to translation, rotation and scale.



Data Reduction

Purpose:

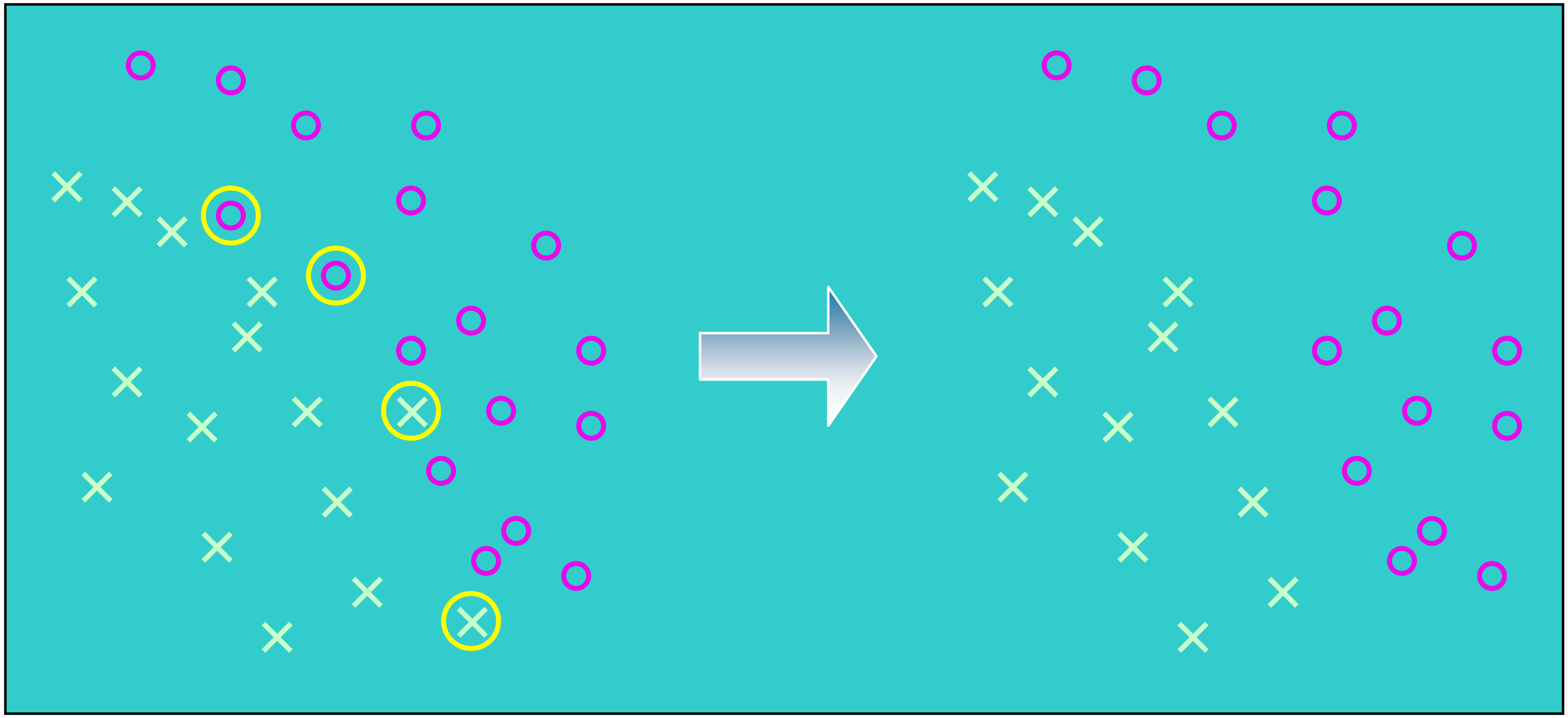
- **Reduce computation load of a classifier**
- **Increase data consistency**

Techniques:

- **To reduce data size:**
  - **Editing**: To eliminate noisy (boundary) data
  - **Condensing**: To eliminate redundant (deeply embedded) data
  - **Vector quantization**: To find representative data
- **To reduce data dimensions:**
  - **Principal component analysis**: To reduce the dimensions of the feature sets
  - **Discriminant analysis**: To find the best set of vectors which best separates the patterns

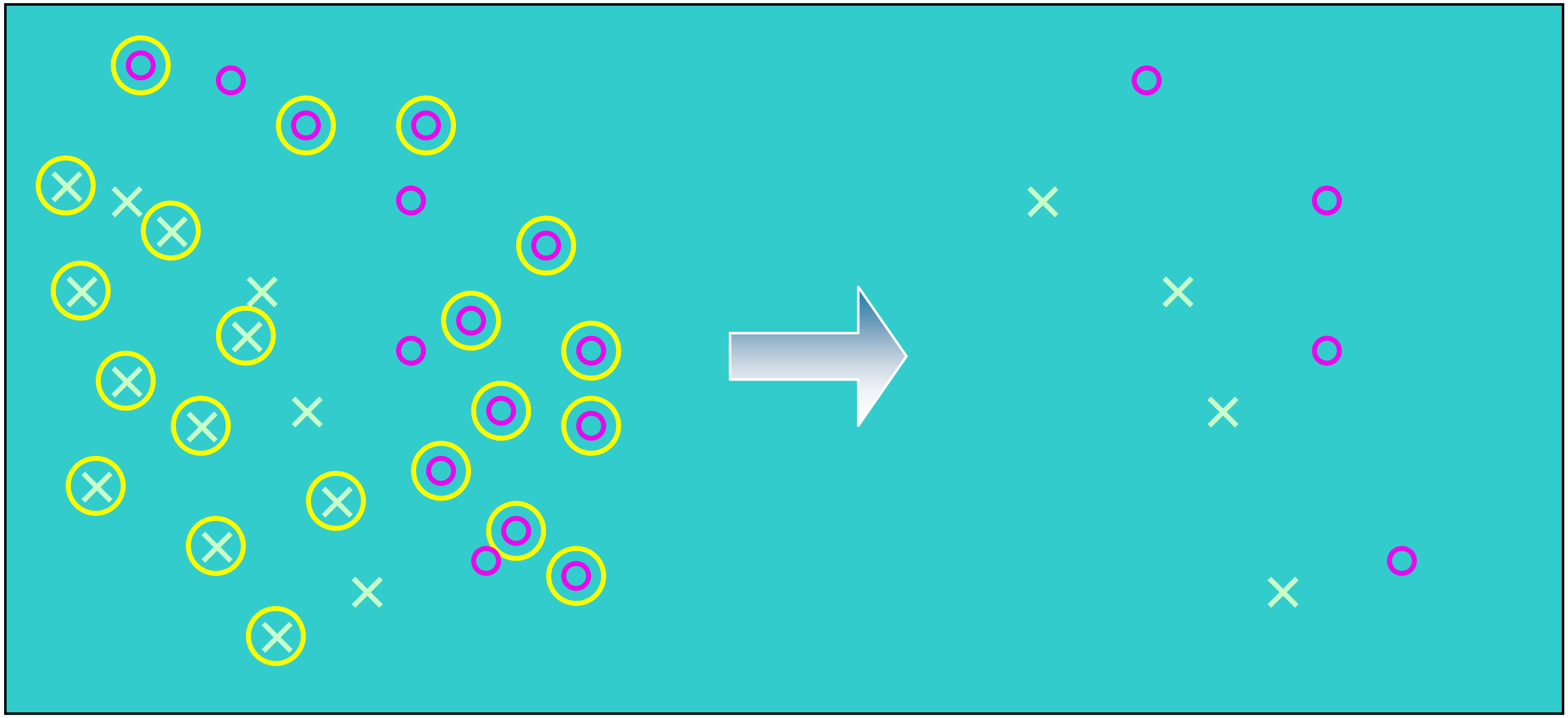
Data Editing

To remove noisy (boundary) data



Data Condensing

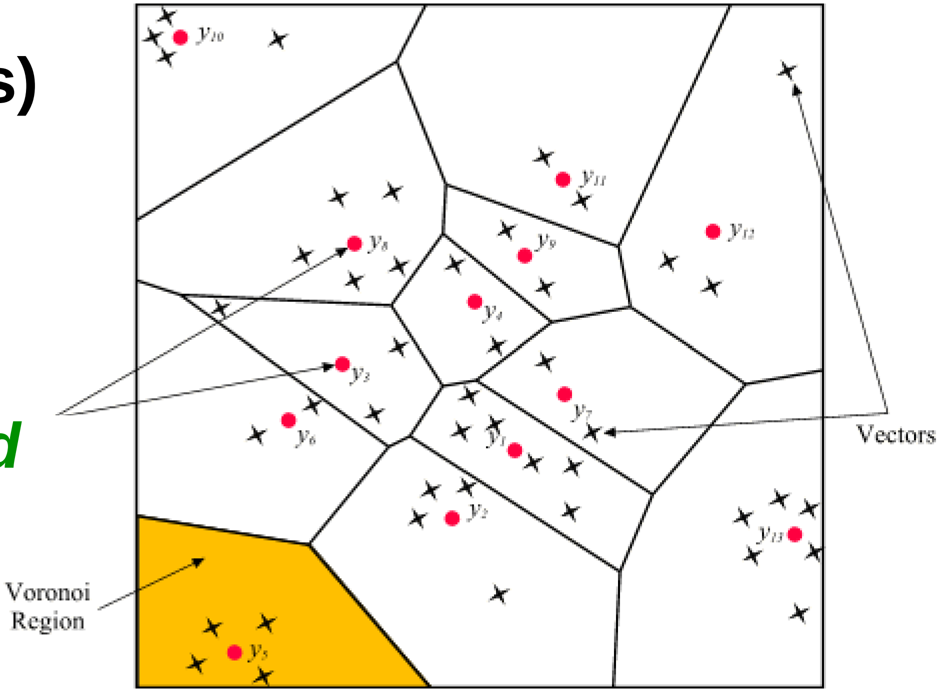
To remove redundant (deeply embedded) data



Vector quantization

- Modeling of probability density functions by the distribution of prototype vectors

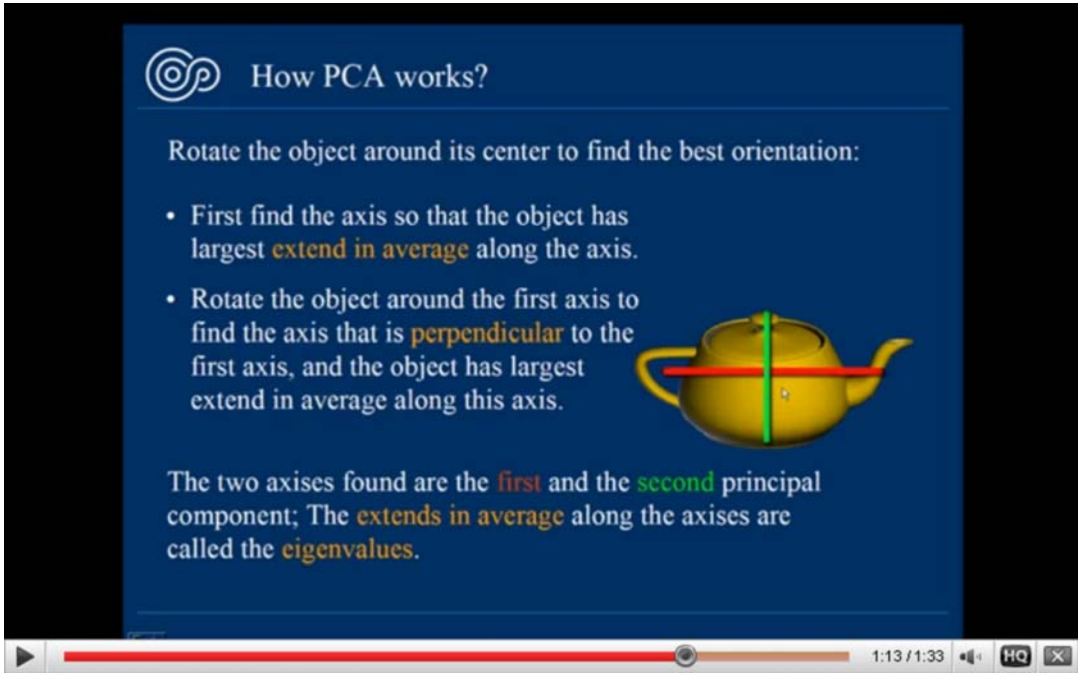
- It works by dividing a large set of points (vectors) into *groups* having approximately the same number of points closest to them. Each group is represented by its *centroid* point, as in k-means and some other clustering algorithms



A Layman's Introduction to Principal Component Analysis

Projection onto eigenvectors that correspond to the first few largest eigenvalues of the covariance matrix

http://www.youtube.com/watch?v=BfTMmoDFXyE&feature=related



Principal Component Analysis

- In PCA, we seek to represent the d-dimensional data in a lower-dimensional space. This:
  - Reduces the degrees of freedom
  - Reduces the space and time complexities

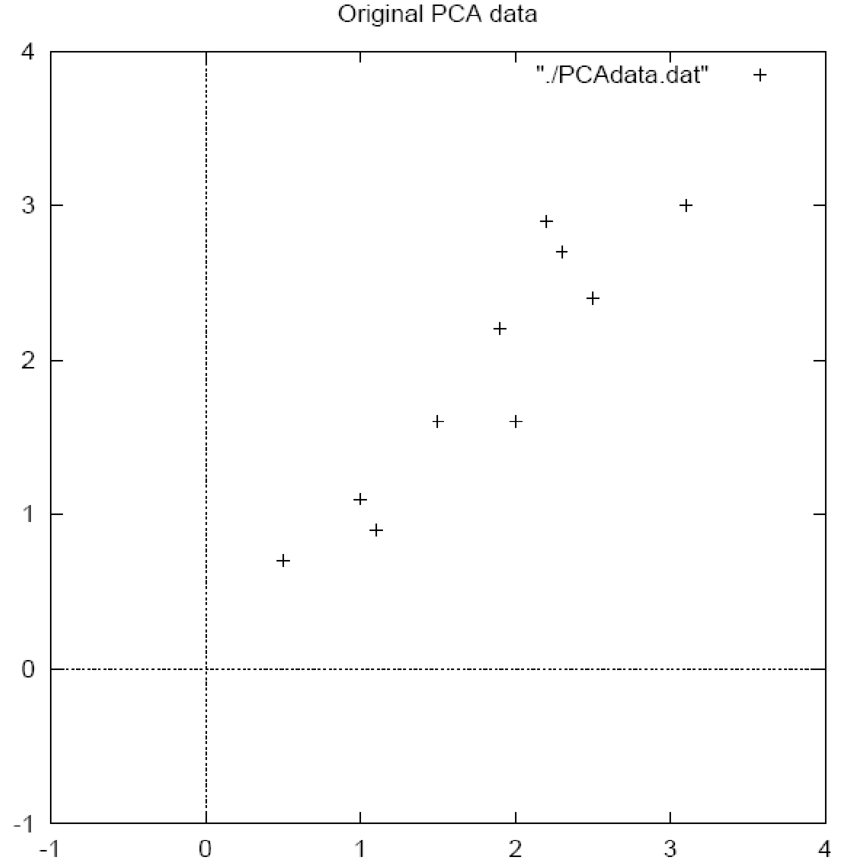
- The goal is to represent data in a space that best describes the variation in a sum-squared error sense

Principal Component Analysis

Step 1: Get some data

Data =

x	y
2.5	2.4
0.5	0.7
2.2	2.9
1.9	2.2
3.1	3.0
2.3	2.7
2	1.6
1	1.1
1.5	1.6
1.1	0.9



Principal Component Analysis

Step 2: Subtract the mean

- From each of the data dimensions (from x dimension and y dimension)

Data =

x	y
2.5	2.4
0.5	0.7
2.2	2.9
1.9	2.2
3.1	3.0
2.3	2.7
2	1.6
1	1.1
1.5	1.6
1.1	0.9

DataAdjust =

x	y
.69	.49
-1.31	-1.21
.39	.99
.09	.29
1.29	1.09
.49	.79
.19	-.31
-.81	-.81
-.31	-.31
-.71	-1.01

Principal Component Analysis

Step 3: Calculate the covariance matrix

Data =

x	y
2.5	2.4
0.5	0.7
2.2	2.9
1.9	2.2
3.1	3.0
2.3	2.7
2	1.6
1	1.1
1.5	1.6
1.1	0.9

$$cov = \begin{pmatrix} .616555556 & .615444444 \\ .615444444 & .716555556 \end{pmatrix}$$

Principal Component Analysis

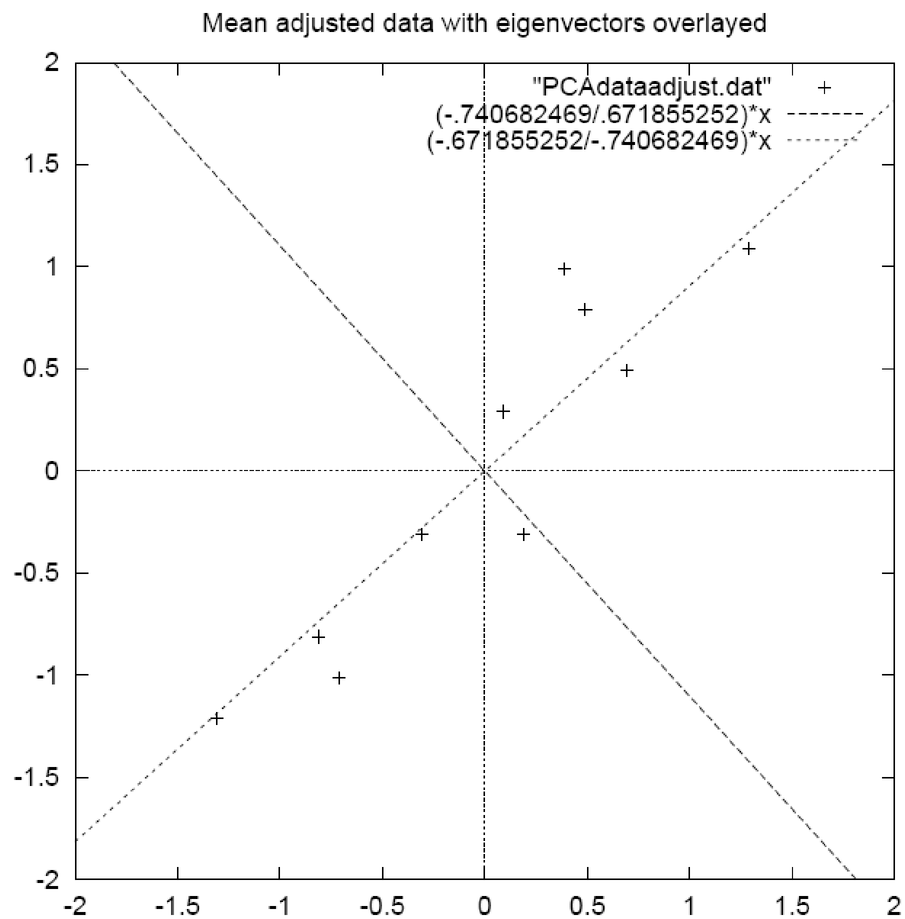
Step 4: Calculate the unit eigenvectors and eigenvalues of the covariance matrix

$$cov = \begin{pmatrix} .616555556 & .615444444 \\ .615444444 & .716555556 \end{pmatrix}$$

$$\Downarrow$$

$$eigenvalues = \begin{pmatrix} .0490833989 \\ 1.28402771 \end{pmatrix}$$

$$eigenvectors = \begin{pmatrix} -.735178656 & -.677873399 \\ .677873399 & -.735178656 \end{pmatrix}$$



Principal Component Analysis

- The 1st eigenvector (*principle component*) shows how data in two dimensions are related along the eigenvector line
- The 2nd eigenvector shows that all the points are off to the side of the main line by some amount

- Eigenvectors are lines that characterize the data
- The next steps: transforming the data so that it is expressed in terms of these lines

Principal Component Analysis

- **Step 5: Choosing components and forming a feature vector**
  - **Order eigenvectors by eigenvalues**
    - This gives the components in order of significance
    - You can decide to ignore the components of lesser significance → final data will have less dimensions (p < d)
  - **Form a feature vector (matrix of vectors)**
    - Take the iegenvectors that you want to keep, and form matrix with these eigenvectors in the columns

$$FeatureVector = (eig_1 eig_2 ... eig_n)$$

Principal Component Analysis

- For our example, two feature vectors are possible:

$$eigenvalues = \begin{pmatrix} .0490833989 \\ 1.28402771 \end{pmatrix}$$

$$eigenvectors = \begin{pmatrix} -.735178656 & -.677873399 \\ .677873399 & -.735178656 \end{pmatrix}$$

$$\begin{pmatrix} -.677873399 & -.735178656 \\ -.735178656 & .677873399 \end{pmatrix}$$

or

$$\begin{pmatrix} -.677873399 \\ -.735178656 \end{pmatrix}$$

Principal Component Analysis

Step 6: Deriving the new dataset

$$FinalData = FeatureVector^T \times RowDataAdjust$$

where $RowDataAdjust$ is the mean-adjusted data transposed

- **It will give us the original data solely in terms of the vectors we chose**

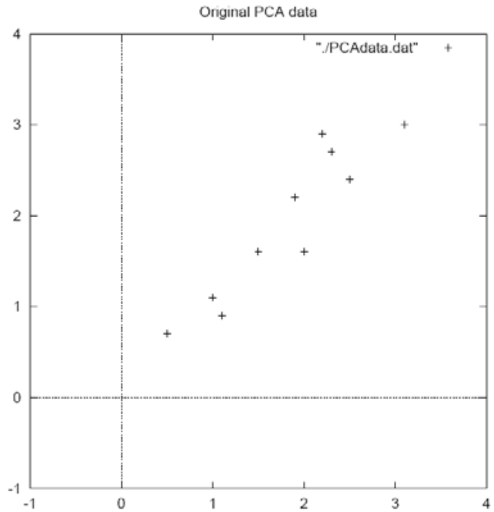
Principal Component Analysis

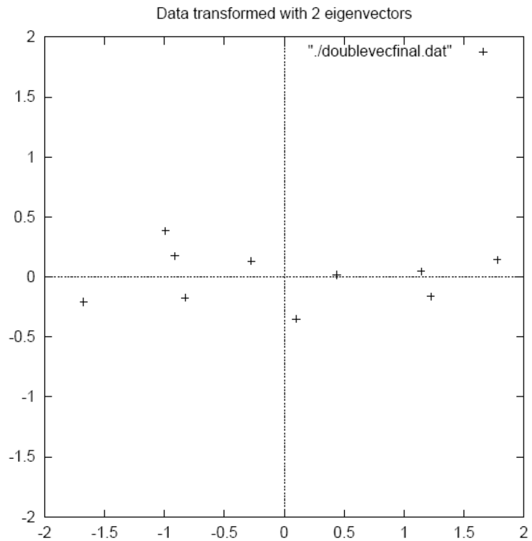
Data =

x	y
2.5	2.4
0.5	0.7
2.2	2.9
1.9	2.2
3.1	3.0
2.3	2.7
2	1.6
1	1.1
1.5	1.6
1.1	0.9

Transformed Data=

x	y
-.827970186	-.175115307
1.77758033	.142857227
-.992197494	.384374989
-.274210416	.130417207
-1.67580142	-.209498461
-.912949103	.175282444
.0991094375	-.349824698
1.14457216	.0464172582
.438046137	.0177646297
1.22382056	-.162675287



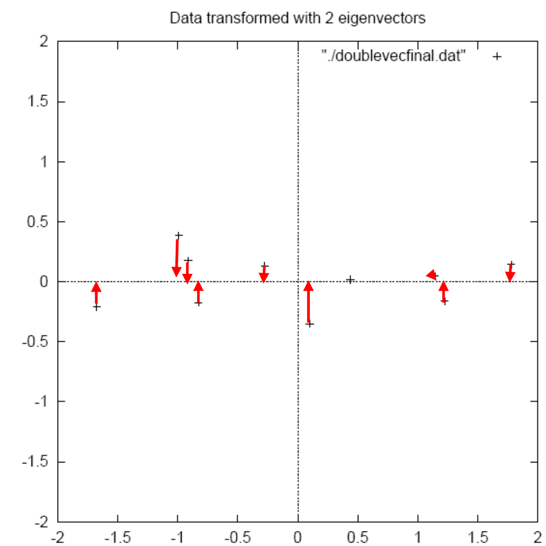


Principal Component Analysis

Transformed Data=

x	y
-.827970186	-.175115307
1.77758033	.142857227
-.992197494	.384374989
-.274210416	.130417207
-1.67580142	-.209498461
-.912949103	.175282444
.0991094375	-.349824698
1.14457216	.0464172582
.438046137	.0177646297
1.22382056	-.162675287

If only one eigenvector was kept, the transformed data will have only one dimension

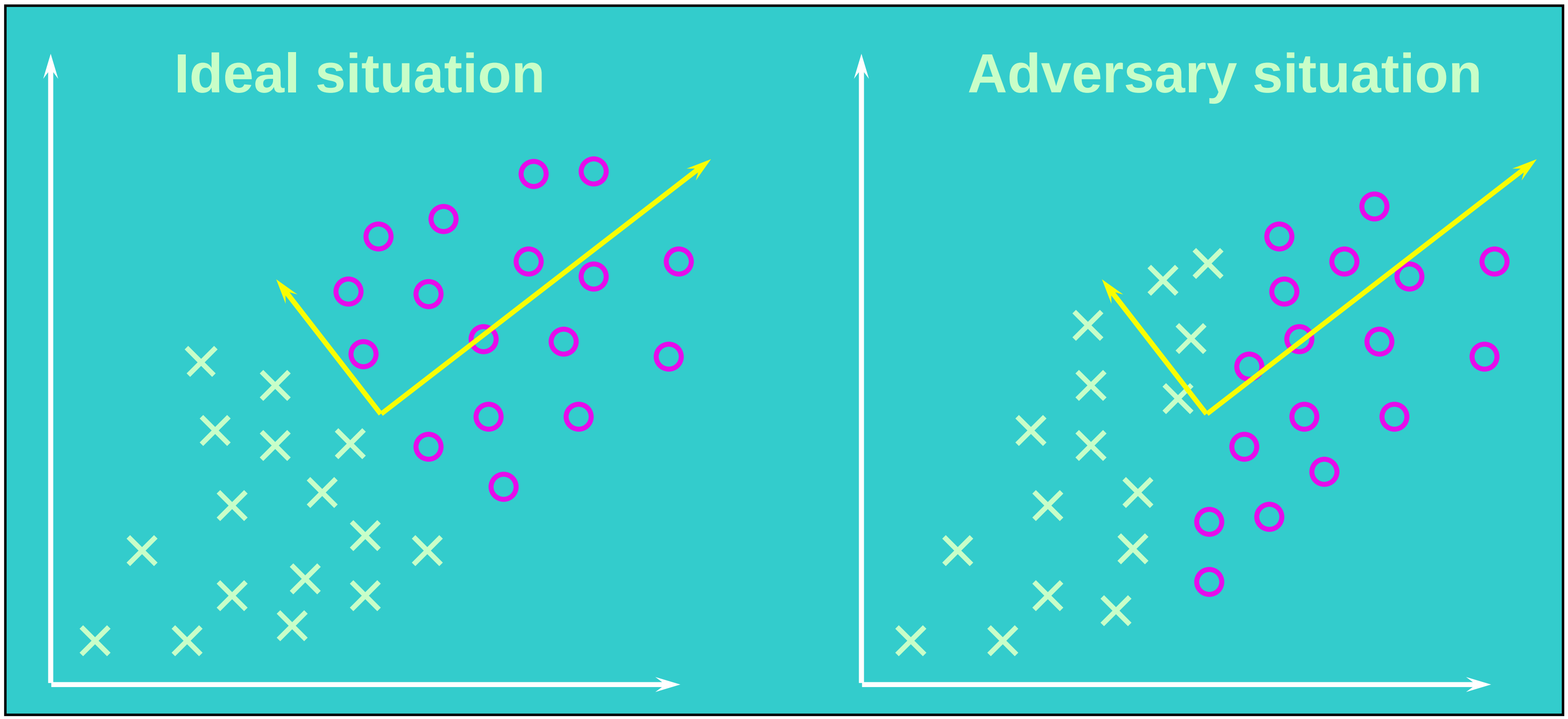


Principal Component Analysis

- **Face recognition**
  - **Original images (say 20) are of peoples faces**
  - **Problem: Given a new image, whose face from the original face is it?**
- **Method**
  - **Measure the difference between the new image and the original images**
    - Not along the original axis, but along the new axes derived from the PCA analysis
    - These axes work much better for recognising faces, because the PCA analysis gives the original images *in terms of the differences and similarities between them*

Principal Component Analysis

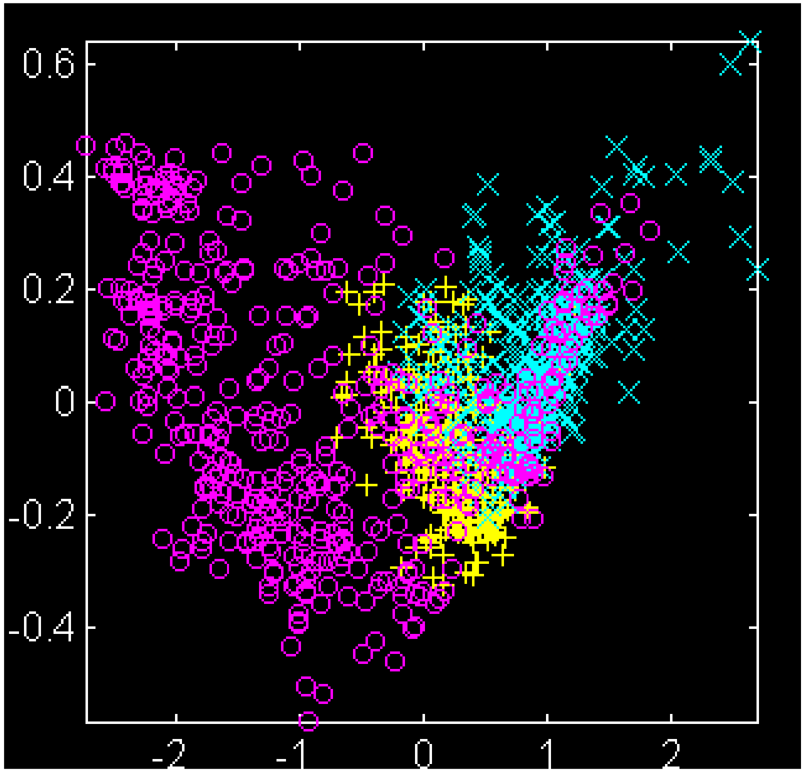
Projection onto eigenvectors that correspond to the first few largest eigenvalues of the covariance matrix



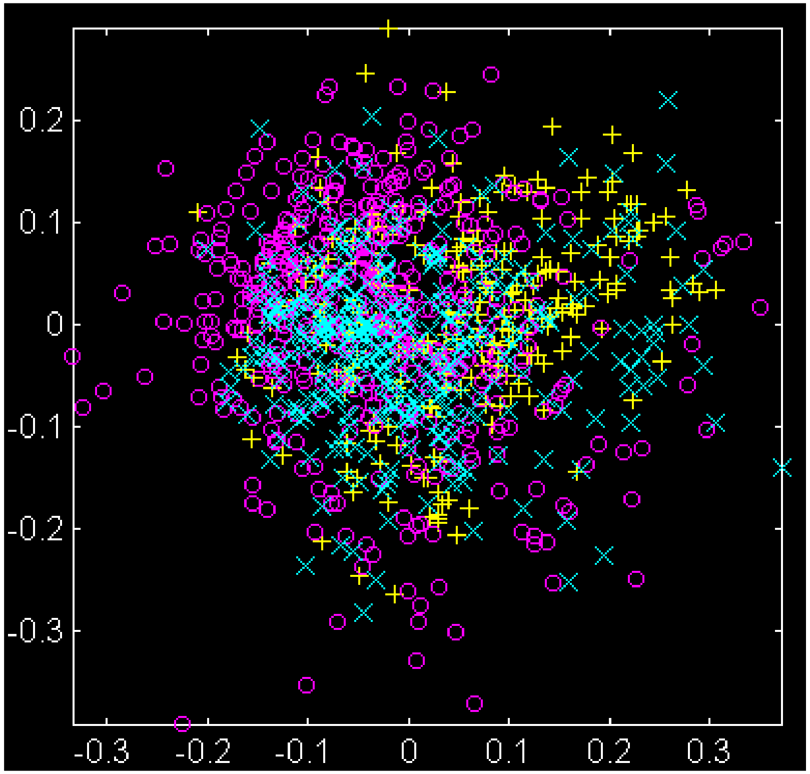
Principal Component Analysis

Eigenvalues of covariance matrix: $\lambda_1 > \lambda_2 > \lambda_3 > ... > \lambda_d$

Projection on v_1 & v_2



Projection on v_3 & v_4



Principal Component Analysis

- How to perform PCA transformation
  - http://lib.stat.cmu.edu/multi/pca.c
  - And many other links with source code available on Internet

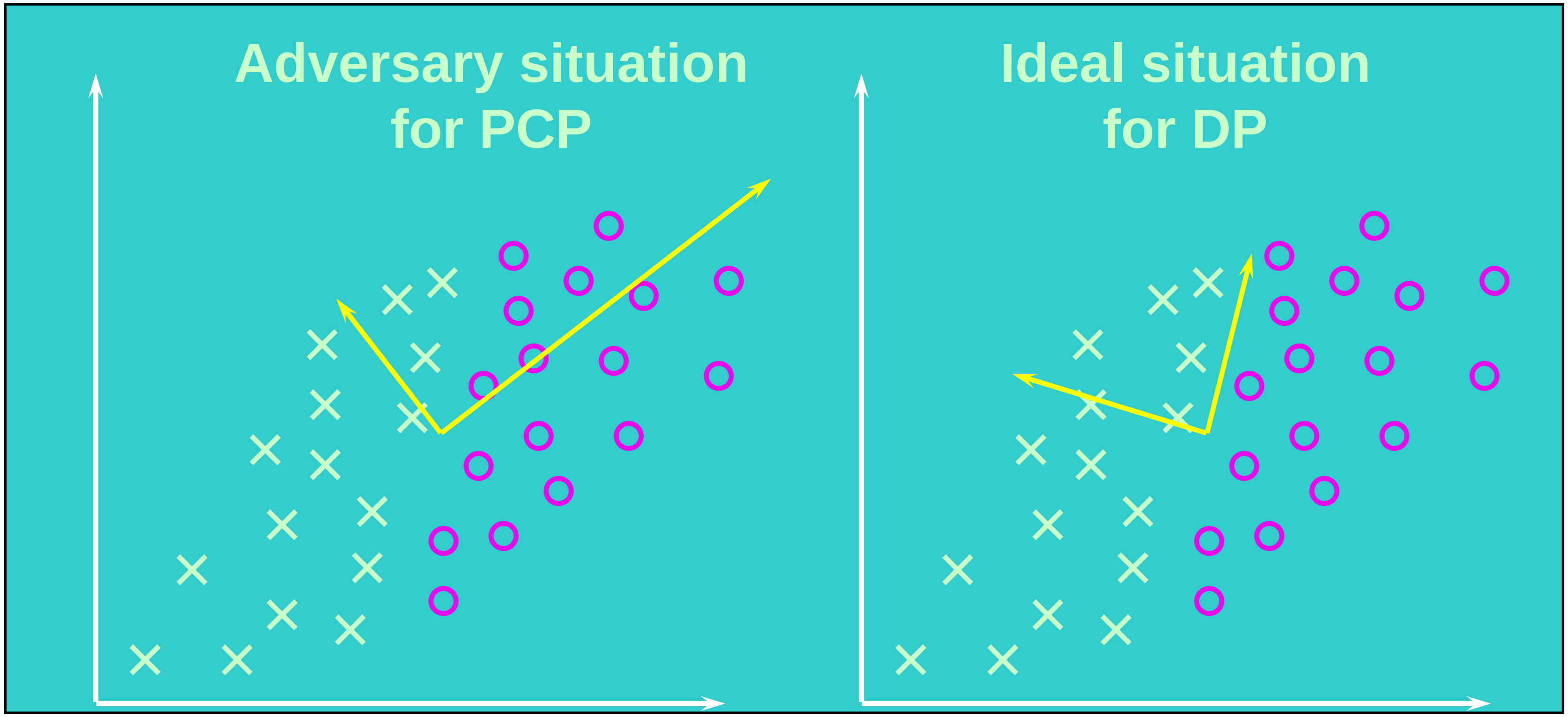
Discriminant Analysis (DA)

- PCA seeks directions that are efficient for *representation*
  - *Unsupervised technique*

- Discriminant analysis seeks directions that are efficient for *discrimination*
  - *Supervised technique*

Discriminant Analysis

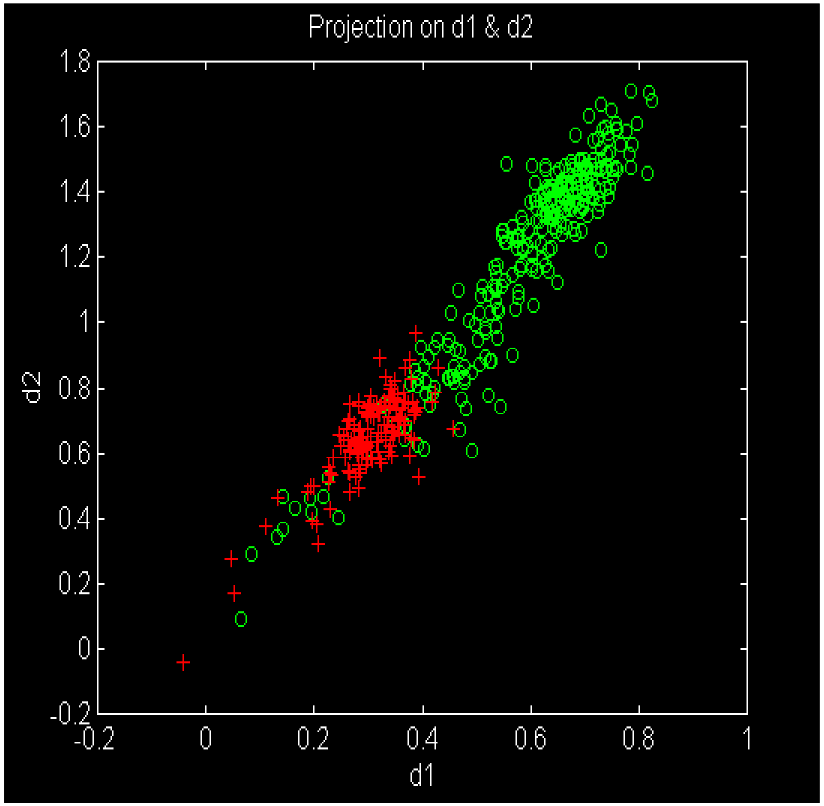
Projection onto directions that can best separate data of different classes



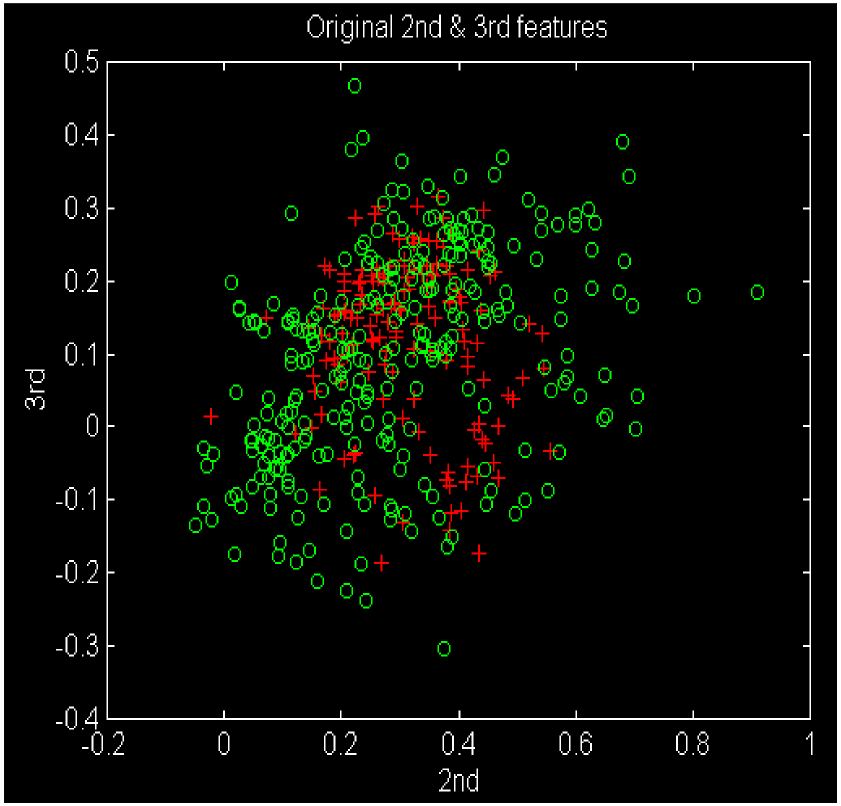
Discriminant Analysis

Best discriminant vectors : $v_1, v_2, \ldots, v_d$

Projection on v_1 & v_2



Projection on v_3 & v_4



Discriminant Analysis

- **Theory of Fisher Linear Discriminant:** **http://www.csd.uwo.ca/~olga/Courses//CS434a_541a//Lecture8.pdf**

- **Demo:** **http://www.inf.ethz.ch/personal/porbanz/ml2/applets/Classifier/JFishersLinearDiscriminantApplet.html**